Atty. Docket No. 166.0001

PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant : Roach *et al.*

Serial No. : 09/957,459

Examiner: To, Baoquoc N.

Filed : September 21, 2001

Group Art Unit: 2172

For : AN AUTOMATIC REAL-TIME FILE MANAGEMENT METHOD AND APPARATUS

DECLARATION UNDER 37 CFR §1.132

I, Steven R. Williams, declare under penalty of perjury, that:

1. I am an electrical engineer and Vice President of precisionWave Corporation with approximately 23 years of experience in the software development industry. I make this declaration based upon my knowledge and years of experience.

2. I earned a Bachelor of Science degree in electrical engineering from the University of Illinois in 1981.

3. I am listed as a co-inventor on U.S. Patent Application No. 09/957,459.

4. I have reviewed U.S. Patent No. 6,629,109 B1, issued to Koshisaka (hereinafter "Koshisaka").

5. Koshisaka teaches "an application-centric" file revision management system for executing file revision management when an application operating on an operating system of a computer system saves a file by file overwrite. The file manipulation monitoring section monitors and hooks Application Program Interface (API) commands outputted by the application to the operating system, thereby detecting file manipulation executed by the application. After a command is hooked, a different instruction is passed to the operating system in order to carry out Koshisaka's revision management system activities. *See* Koshisaka, column 6, lines 32-43.

6. As illustrated by EXHIBIT A, a diagram of a likely implementation of the Koshisaka file revision management system, the system of Koshisaka includes an application 105 that is written for the Koshisaka API, a Koshisaka-specific API 110, an

operating system 115, and a group of user files 120. As Koshisaka is an application-centric system, the file manipulation monitoring section of Koshisaka detects an instruction 125 (for example, a file deletion instruction) which is going to be outputted *by the application*. Koshisaka does not operate by detecting an instruction by an operating system. *See* Koshisaka, column 6, lines 32-43. Thus, Koshisaka fails to teach or suggest at least the "detecting an instruction *by an operating system*" limitation of the claims. Therefore, it would not have been obvious to one of ordinary skill in the art at the time the above-referenced invention was made to modify the teachings of Koshisaka to obtain the above-referenced invention.

7. An alternative implementation of Koshisaka includes an application that is NOT written for a Koshisaka-specific API, but is written to output instructions directly to an operating system via the normal Windows API. Koshisaka suggests the possibility of monitoring such outputted instructions from the application to the operating system, and supplanting these operating system instructions with different operating system instructions in order to carry out Koshisaka's revision management activities. Even in this alternative implementation, Koshisaka cannot be interpreted to teach or suggest at least the "detecting an instruction *by an operating system*" limitation of the claims. As in item 6 above, it would not have been obvious to one of ordinary skill in the art at the time the above-referenced invention was made to modify the teachings of Koshisaka to obtain the above-referenced invention.

8. As a result of detecting the instruction by the application, unlike the invention disclosed in the above-referenced application, Koshisaka provides a very limited layer of file protection.

9. I have reviewed U.S. Patent No. 6,535,894, issued to Schmidt *et al.* (hereinafter "Schmidt").

10. Schmidt discloses an apparatus and method for incremental updating of archive files. An original archive file having one or more entries is created, where each entry in the original archive file is itself a file. The original archive file is transmitted to a client computer, and subsequently, a target archive file is created wherein one or more of its entries are "typically expected" to be identical to one or more entries in the original archive file. A difference file including an index file describing the changes between the

original archive file and the target archive file is also created and transmitted to the client computer. At the client computer, Schmidt teaches that the difference archive file is applied to the original archive file to produce a synthesized archive file. *See* Schmidt, column 9, line 65 – column 10, line 59.

11. Schmidt's field of invention relates to an apparatus and method to facilitate incremental updating of program code. *See* Schmidt, column 1, lines 8-14. Schmidt discloses methods for optimizing transmitted archive files through the creation and manipulation of original archive files, target archive files, difference archive files and synthesized archive files. Schmidt teaches improvements to the deployment of program code from server computers to client computers.

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

Executed this 7th day of September 2004.

Steven R. Williams